IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Patent Application of	)	
	)	
Sanjay GHEMAWAT et al.	)	**ATTN: Appeal Brief - Patents**
	)	
Application No.: 10/608,037	)	Group Art Unit: 2167
	)	
Filed: June 30, 2003	)	Examiner: K. Lovel
	)	
For: MAINTAINING DATA IN A FILE SYSTEM	)	
	)	

U.S. Patent and Trademark Office
Customer Window, Mail Stop Appeal Brief - Patents
Randolph Building
401 Dulany Street
Alexandria, VA 22314

APPEAL BRIEF

This Appeal Brief is submitted in response to the final Office Action, dated December 10, 2008, and in support of the Notice of Appeal, filed March 10, 2009.

I. REAL PARTY IN INTEREST

The real party in interest in this appeal is Google Inc.

II. RELATED APPEALS, INTERFERENCES, AND JUDICIAL PROCEEDINGS

Appellants are unaware of any related appeals, interferences, or judicial proceedings.

III. STATUS OF CLAIMS

Claims 1, 3-11, 13, 15, 16, and 19-39 are pending in this application.

Claims 1, 3-11, 13, 15, 16, 19-24, 27, 30, 31, and 34-39 have been finally rejected under 35 U.S.C. § 103(a) as unpatentable over Dinker et al. (U.S. Patent No. 7,206,836) in view of Bobbitt et al. (U.S. Patent Application Publication No. 2003/0115218).

Claims 25, 26, 28, 29, 32, and 33 have been finally rejected under 35 U.S.C. § 103(a) as unpatentable over Dinker et al. in view of Bobbitt et al. and Rao et al. (U.S. Patent No. 5,689,706).

Claims 2, 12, 14, 17, and 18 were previously canceled without prejudice or disclaimer.

Claims 1, 3-11, 13, 15, 16, and 19-39 are the subject of the present appeal. These claims are reproduced in the Claim Appendix of this Appeal Brief.

IV. STATUS OF AMENDMENTS

No Amendments were filed subsequent to the final Office Action. Appellants filed a Request for Reconsideration subsequent to the final Office Action, which resulted in the issuance of an Advisory Action.

V. SUMMARY OF CLAIMED SUBJECT MATTER

In the paragraphs that follow, a concise explanation of the independent claims, the dependent claims argued separately, and the claims reciting means-plus-function or step-plus-function language that are involved in this appeal will be provided by referring, in parenthesis, to examples of where support can be found in the specification and drawings.

Claim 1 recites a file system (e.g., Fig. 1), comprising: a plurality of servers (e.g., Fig. 1, 120) configured to store file data as chunks (e.g., page 6, lines 14-15); and a master (e.g., Fig. 1, 130) connected to the servers (e.g., Fig. 1; page 5, lines 17-20) and configured to store namespace data that includes file identifiers for files for which the file data is stored as chunks (e.g., Fig. 4, 410; page 9, lines 17-21), store mapping data that maps the file identifiers to the chunks to which the file identifiers correspond (e.g., Fig. 4, 420; page 10, lines 1-3), store an operation log that includes a record of changes to at least one of the namespace data or the mapping data (e.g., Fig. 4, 440; page 11, lines 7-13), and store location data that identifies which of the servers stores which of the chunks (e.g., Fig. 4, 430; page 10, lines 10-22), where the master is configured to: communicate with the servers at startup of the master to identify the chunks stored by the servers (e.g., page 10, lines 10-14), and record, in a non-persistent manner, information regarding the chunks stored by each of the servers as the location data (e.g., page 10, lines 10-12).

Claim 11 recites that the information includes version numbers of the chunks (e.g., page 10, lines 8-9 and 19-22).

Claim 13 recites a master (e.g., Fig. 1, 130) in a file system (e.g., Fig. 1) that includes the master connected to a plurality of servers (e.g., Fig. 1, 120). The master comprises means for communicating with the servers to identify file data stored by the servers as chunks (e.g., Fig. 3, 320, 380; page 10, lines 12-22); means for storing, in a non-persistent manner, location information that identifies ones of the servers that store the chunks (e.g., Fig. 3, 320, 330; page 10, lines 10-12); means for updating the location information by periodically instructing the servers to identify the data stored by the servers (e.g., Fig. 3, 320, 380; page 10, lines 16-22);

means for storing namespace data that includes file identifiers for files for which the file data is stored as chunks by the servers (e.g., Fig. 3, 320, 350; page 9, lines 17-21); means for storing mapping data that maps the file identifiers to the chunks to which the file identifiers correspond (e.g., Fig. 3, 320, 350; page 10, lines 1-3); and means for storing an operation log that includes a record of changes to at least one of the namespace data or the mapping data (e.g., Fig. 3, 320, 350; page 11, lines 7-13).

Claim 15 recites a file system (e.g., Fig. 1) that comprises a plurality of servers (e.g., Fig. 1, 120) configured to store files as chunks (e.g., page 6, lines 14-15); and a master (e.g., Fig. 1, 130) connected to the servers (e.g., Fig. 1; page 5, lines 17-20) and configured to store namespace data that includes file identifiers for which the files are stored as chunks (e.g., Fig. 4, 410; page 9, lines 17-21), store mapping data that maps the file identifiers to the chunks to which the file identifiers correspond (e.g., Fig. 4, 420; page 10, lines 1-3), store an operation log that includes a record of changes to at least one of the namespace data or the mapping data (e.g., Fig. 4, 440; page 11, lines 7-13), and store location data that identifies which of the servers stores which of the chunks (e.g., Fig. 4, 430; page 10, lines 10-22), where the master is configured to determine location information by communicating with the servers (e.g., page 10, lines 10-14), the location information being based on which of the servers store ones of the chunks (e.g., page 10, lines 10-14), store the location information as the location data (e.g., page 10, lines 10-14), and update the location data by periodically communicating with the servers to obtain changes to the location data (e.g., page 10, lines 16-22).

Claim 16 recites a file system (e.g., Fig. 1) that comprises a plurality of servers (e.g., Fig. 1, 120) configured to store file data as chunks (e.g., page 6, lines 14-15); and a master (e.g., Fig.

1, 130) connected to the servers (e.g., Fig. 1; page 5, lines 17-20) and configured to store namespace data that includes file identifiers for files for which the file data is stored as chunks (e.g., Fig. 4, 410; page 9, lines 17-21), store mapping data that maps the file identifiers to the chunks to which the file identifiers correspond (e.g., Fig. 4, 420; page 10, lines 1-3), store an operation log that includes a record of changes to the namespace data and the mapping data (e.g., Fig. 4, 440; page 11, lines 7-13), and store location data that identifies which of the servers stores which of the chunks (e.g., Fig. 4, 430; page 10, lines 10-22), where the master is configured to communicate with the servers to determine location information of the data (e.g., page 10, lines 10-22), the location information being based on which of the servers store the chunks (e.g., page 10, lines 10-14), and store the location information as the location data (e.g., page 10, lines 10-14).

Claim 20 recites that the master stores the namespace data using prefix-compression (e.g., page 9, lines 19-21).

Claim 22 recites that the chunk handle encodes a timestamp (e.g., page 10, lines 2-9).

Claim 23 recites that the master is configured to update the location data by periodically instructing the servers to provide information regarding the chunks stored by the servers (e.g., page 10, lines 16-22).

Claim 24 recites that the operation log includes a logical timeline that defines an order for concurrent operations (e.g., page 11, lines 7-11).

Claim 25 recites that the master is configured to determine when a size of the operation log exceeds a threshold (e.g., page 11, lines 19-21), and create a checkpoint of the operation log when the size of the operation log exceeds the threshold (e.g., page 11, lines 19-21).

Claim 26 recites that the master is configured to create a new operation log file (e.g., page 11, lines 11-13; page 12, lines 8-11), and create the checkpoint as a background operation (e.g., page 12, lines 8-11).

Claim 27 recites that the operation log includes a logical timeline that defines an order for concurrent operations (e.g., page 11, lines 7-11).

Claim 28 recites means for determining when a size of the operation log exceeds a threshold (e.g., page 11, lines 19-21); and means for creating a checkpoint of the operation log when the size of the operation log exceeds the threshold (e.g., page 11, lines 19-21).

Claim 29 recites that the means for creating the checkpoint includes means for creating a new operation log file (e.g., Fig. 3, 320; page 11, lines 11-13; page 12, lines 8-11), and means for creating the checkpoint as a background operation (e.g., Fig. 3, 320; page 12, lines 8-11).

Claim 30 recites a method performed by a master device (e.g., Fig. 1, 130) in a file system (e.g., Fig. 1) that includes the master device connected to a plurality of server devices (e.g., Fig. 1, 120). The method comprises communicating with the server devices to identify file data stored by the server devices as chunks (e.g., page 10, lines 10-22); storing location information that identifies ones of the server devices that store the chunks (e.g., page 10, lines 10-14); storing namespace data that includes file identifiers for files for which the file data is stored as chunks by the server devices (e.g., page 9, lines 17-21); storing mapping data that maps the file identifiers to the chunks to which the file identifiers correspond (e.g., page 10, lines 1-3); and maintaining an operation log that includes a record of changes to the namespace data and the mapping data (e.g., page 11, lines 7-13).

Claim 31 recites that maintaining the operation log includes storing a logical timeline that

defines an order for operations including concurrent operations (e.g., page 11, lines 7-11).

Claim 32 recites determining when a size of the operation log exceeds a threshold (e.g., page 11, lines 19-21); and creating a checkpoint of the operation log when the size of the operation log exceeds the threshold (e.g., page 11, lines 19-21).

Claim 33 recites that creating the checkpoint includes creating a new operation log file (e.g., page 11, lines 11-13; page 12, lines 8-11), and creating the checkpoint as a background operation (e.g., page 12, lines 8-11).

Claim 35 recites that the master is further configured to identify one or more of the servers to store a new chunk based on failure correlation properties associated with the servers (e.g., page 15, line 20 – page 16, line 8), and place the new chunk at the identified one or more servers (e.g., page 15, lines 5-8; page 16, lines 9-11).

Claim 37 recites that the master is further configured to identify one or more of the servers to store a new chunk based on failure correlation properties associated with the servers (e.g., page 15, line 20 – page 16, line 8), and place the new chunk at the identified one or more servers (e.g., page 15, lines 5-8; page 16, lines 9-11).

Claim 38 recites means for identifying one or more of the servers to store a new chunk based on utilization of the servers (e.g., page 15, lines 8-12), prior chunk distribution involving the servers (e.g., page 15, lines 13-19), and failure correlation properties associated with the servers (e.g., page 15, line 20 – page 16, line 8); and means for placing the new chunk at the identified one or more servers (e.g., page 15, lines 5-8; page 16, lines 9-20).

Claim 39 recites identifying one or more of the server devices to store a new chunk based on utilization of the server devices (e.g., page 15, lines 8-12), prior chunk distribution involving

the server devices (e.g., page 15, lines 13-19), and failure correlation properties associated with the server devices (e.g., page 15, line 20 – page 16, line 8); and placing the new chunk at the identified one or more server devices (e.g., page 15, lines 5-8; page 16, lines 9-20).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

A. Claims 1, 3-11, 13, 15, 16, 19-24, 27, 30, 31, and 34-39 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Dinker et al. in view of Bobbitt et al.

B. Claims 25, 26, 28, 29, 32, and 33 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Dinker et al. in view of Bobbitt et al. and Rao et al.

VII. ARGUMENT

A. The Rejection Under 35 U.S.C. § 103(a) Based on Dinker et al. in View of Bobbitt et al. Should be Reversed.

The initial burden of establishing a prima facie basis to deny patentability to a claimed invention is always upon the Examiner. In re Oetiker, 977 F.2d 1443, 24 USPQ2d 1443 (Fed. Cir. 1992). In rejecting a claim under 35 U.S.C. § 103, the Examiner must provide a factual basis to support the conclusion of obviousness. In re Warner, 379 F.2d 1011, 154 USPQ 173 (CCPA 1967). Based upon the objective evidence of record, the Examiner is required to make the factual inquiries mandated by Graham v. John Deere Co., 86 S.Ct. 684, 383 U.S. 1, 148 USPQ 459 (1966). KSR International Co. v. Teleflex Inc., 550 U.S. _____ (April 30, 2007). The Examiner is also required to explain how and why one having ordinary skill in the art would have been led to modify an applied reference and/or combine applied references to arrive at the

claimed invention. Uniroyal, Inc. v. Rudkin-Wiley Corp., 837 F.2d 1044, 5 USPQ2d 1434 (Fed. Cir. 1988).

1. Claims 1, 3-10, 19, and 21.

Independent claim 1 is directed to a file system that comprises a plurality of servers configured to store file data as chunks; and a master connected to the servers and configured to store namespace data that includes file identifiers for files for which the file data is stored as chunks, store mapping data that maps the file identifiers to the chunks to which the file identifiers correspond, store an operation log that includes a record of changes to at least one of the namespace data or the mapping data, and store location data that identifies which of the servers stores which of the chunks, where the master is configured to communicate with the servers at startup of the master to identify the chunks stored by the servers, and record, in a non-persistent manner, information regarding the chunks stored by each of the servers as the location data.

Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, do not disclose or suggest the combination of features recited in claim 1. For example, Dinker et al. and Bobbitt et al. do not disclose or suggest a master that is configured to, among other things, store an operation log that includes a record of changes to at least one of namespace data, which includes file identifiers for files for which the file data is stored as chunks, or mapping data, which maps the file identifiers to the chunks to which the file identifiers correspond, as recited in claim 1.

The Examiner admitted that Dinker et al. does not disclose or suggest an operation log, and alleged that Bobbitt et al. discloses storing an operation log that includes a record of changes

to at least one of namespace data or mapping data, and cited paragraphs 0048 and 0052-0054 of Bobbitt et al. for support. Final Office Action, pages 3-4. Appellants submit that the disclosure of Bobbitt et al. provides no support for the Examiner's allegation.

At paragraph 0048, Bobbitt et al. discloses:

> Configuration information 45 includes configuration data that identifies what physical server(s) the various gtrees for a given GVV are hosted on, what physical devices the master and slave gtrees are stored on, the exports each server provides, and the roles played by the various components in a Gossamer virtual file system. Configuration information also may include schedule data (i.e., data pertaining to when migrations are to be performed or considered, when backups are to occur, when the background consistency checker may run, etc.), status files pertaining to operations in progress, such as migration and backup operations, and log files. The configuration information may be stored on one of the servers used to store the master gtrees and/or the slave gtrees, including file server 20, or may be stored on a separate server that is not used to store file system data files that are part of a GVV.

In this section, Bobbitt et al. discloses configuration information that identifies what server hosts various gtrees, what devices store the master and slave gtrees, the exports provided by each server, the roles the various components play, schedule data, status files pertaining to operations in progress, and log files. While this section of Bobbitt et al. mentions "log files," nowhere does Bobbitt et al. disclose or remotely suggest that these log files include a record of changes to namespace data that includes file identifiers for files for which the file data is stored as chunks. Bobbitt et al. also does not disclose or remotely suggest that these log files include a record of changes to mapping data that maps the file identifiers to the chunks to which the file identifiers correspond. Thus, Bobbitt et al. does not disclose or suggest a master that is configured to, among other things, store an operation log that includes a record of changes to at least one of namespace data, which includes file identifiers for files for which the file data is stored as

chunks, or mapping data, which maps the file identifiers to the chunks to which the file identifiers correspond, as recited in claim 1.

At paragraphs 0052-0054, Bobbitt et al. discloses a user-view tree that corresponds to a virtual directory and file hierarchy. Bobbitt et al. discloses that translations between virtual pathname and actual server-pathname are handled through a GVV master directory structure that logically divides its data into three spaces: a Gossamer namespace, a temporary migrating space, and a garbage space. Nowhere does Bobbitt et al. disclose or suggest anything that can reasonably correspond to an operation log that includes a record of changes to namespace data and/or mapping data. Thus, Bobbitt et al. does not disclose or suggest a master that is configured to, among other things, store an operation log that includes a record of changes to at least one of namespace data, which includes file identifiers for files for which the file data is stored as chunks, or mapping data, which maps the file identifiers to the chunks to which the file identifiers correspond, as recited in claim 1.

In response to these arguments, when previously presented by Appellants, the Examiner alleged that an example of a namespace is a directory and the gtrees of Bobbitt et al. are considered to represent a directory. Final Office Action, page 16. The Examiner further alleged that Bobbitt et al. deals with the migration of files, which means that files are being moved from one server to another server, and when files are migrated, their namespace data and mapping data are both updated at the master. Final Office Action, page 16. The Examiner alleged that this updating at the master corresponds to an operation log. Final Office Action, page 16. Appellants submit that the Examiner's allegations lack merit.

Even assuming, for the sake of argument, that the gtrees, disclosed by Bobbitt et al., can

reasonably be interpreted as including namespace data and mapping data, as alleged by the Examiner (a point that Appellants do not concede), the migration of files would not result in the updating of the namespace data and the mapping data, as alleged by the Examiner. As recited in claim 1, namespace data includes file identifiers for files for which file data is stored as chunks. Contrary to the Examiner's allegation, migrating a file would have no bearing on the namespace data. The migrated file would continue to exist in the file system and only its location would change. Thus, the namespace data corresponding to the file would remain unchanged. As recited in claim 1, mapping data maps file identifiers to the chunks to which the file identifiers correspond. Contrary to the Examiner's allegation, migrating a file would have no bearing on the mapping data. The migrated file (if stored as chunks) would continue to be mapped to the same chunks, though the location of one or more of the chunks might change. Thus, the mapping data corresponding to the file would remain unchanged. For at least these reasons, Appellants submit that the Examiner's allegations lack merit.

Further, even assuming, for the sake of argument, that the disclosure of Bobbitt et al. could reasonably be interpreted as disclosing updating namespace data and mapping data when a file is migrated, as alleged by the Examiner (a point that Appellants do not concede), Bobbitt et al. still would not disclose an operation log that includes a record of changes to at least one of the namespace data or the mapping data. Rather, under the Examiner's allegation, namespace data and mapping data would be updated when a file is migrated. Simply updating namespace data and mapping data is a very different function with a completely different result from storing a record of changes to at least one of the namespace data or the mapping data. Thus, Bobbitt et al. does not disclose or suggest a master that is configured to, among other things, store an

operation log that includes a record of changes to at least one of namespace data, which includes file identifiers for files for which the file data is stored as chunks, or mapping data, which maps the file identifiers to the chunks to which the file identifiers correspond, as recited in claim 1.

In the Advisory Action, the Examiner alleged:

> Paragraph [0048] of Bobbit states that the configuration information of the file system includes log files. It is well-known in the art to one of ordinary skill in the art that a log file maintains a history of actions taken. Further, paragraph [0094] discloses appending the GUID for the destination tree to the end of a pointer tree. Therefore, since the new record is appended to the old, the changes are being maintained. Furthermore, the examiner respectfully disagrees that the changes being made are not to namespace data wherein the namespace data includes file identifiers for files which the data is stored. According to [0086], when a new data file is added to a particular directory, the file name is added to the directory in the master gtree. According to [0042] of Appellant's specification, namespace data can include names of files and the files may be organized hierarchically in a tree of directories and identified by pathname. Therefore, the gtrees of Bobbit are considered to meet the requirements of the defined namespace data since the gtrees include the required information.

Advisory Action, page 2. Appellants submit that the Examiner's allegations lack merit. The Examiner's allegations will be addressed one at a time.

The Examiner alleged:

> Paragraph [0048] of Bobbit states that the configuration information of the file system includes log files. It is well-known in the art to one of ordinary skill in the art that a log file maintains a history of actions taken.

Advisory Action, page 2. Even assuming, for the sake of argument, that the Examiner is correct that a log file maintains a history of actions taken (a point that Appellants do not concede), the Examiner has not established that these "actions" correspond to changes to at least one of namespace data, which includes file identifiers for files for which the file data is stored as chunks, or mapping data, which maps the file identifiers to the chunks to which the file

identifiers correspond, as recited in claim 1. Even though Bobbitt et al. discloses "log files," Bobbitt et al. does not disclose or suggest that these "log files" include a record of changes to at least one of namespace data or mapping data. Thus, Bobbitt et al. does not disclose or suggest a master that is configured to, among other things, store an operation log that includes a record of changes to at least one of namespace data, which includes file identifiers for files for which the file data is stored as chunks, or mapping data, which maps the file identifiers to the chunks to which the file identifiers correspond, as recited in claim 1.

The Examiner alleged:

> Further, paragraph [0094] discloses appending the GUID for the destination tree to the end of a pointer tree. Therefore, since the new record is appended to the old, the changes are being maintained.

Advisory Action, page 2. The Examiner is misconstruing the disclosure of Bobbitt et al. in a failed attempt to reconstruct the claimed invention using impermissible hindsight.

Bobbitt et al. discloses a master gtree that stores directories and their contents and attributes, and slave gtrees that store files and their contents and attributes. Bobbitt et al., paragraph 0050. Bobbitt et al. discloses that the master and slave gtrees are connected by file pointers, which are objects on the master gtree that map from the files virtual pathname to a globally unique identifier (GUID) for the file and the gtree that hosts the file. Bobbitt et al., paragraph 0050. In paragraph 0094, Bobbitt et al. discloses that, when a file is migrated, the globally unique identifier for the destination gtree is appended to the pointer file for the data file such that the pointer file includes GUIDname (corresponding to the file to be migrated), GUIDsrc (corresponding to the slave location of the gtree in which the file was originally stored), and GUIDdest (corresponding to the slave location of the gtree in which the file is to be

stored). Contrary to the Examiner's allegation, Bobbitt et al. does not disclose or suggest that the information appended to the pointer file is maintained. Rather, as shown in Figures 12A-12D and described at paragraph 0095 of Bobbitt et al., the GUIDsrc is deleted once the file has been successfully moved. Thus, Bobbitt et al. does not disclose or suggest a master that is configured to, among other things, store an operation log that includes a record of changes to at least one of namespace data, which includes file identifiers for files for which the file data is stored as chunks, or mapping data, which maps the file identifiers to the chunks to which the file identifiers correspond, as recited in claim 1.

The Examiner alleged:

> Furthermore, the examiner respectfully disagrees that the changes being made are not to namespace data wherein the namespace data includes file identifiers for files which the data is stored. According to [0086], when a new data file is added to a particular directory, the file name is added to the directory in the master gtree. According to [0042] of Appellant's specification, namespace data can include names of files and the files may be organized hierarchically in a tree of directories and identified by pathname. Therefore, the gtrees of Bobbit are considered to meet the requirements of the defined namespace data since the gtrees include the required information.

Advisory Action, page 2. Even assuming, for the sake of argument, that a gtree, disclosed by Bobbitt et al., can reasonably correspond to namespace data (a point that Appellants do not concede), Bobbitt et al. does not disclose or suggest an operation log that includes a record of changes to the gtree. The Examiner has provided absolutely no evidence to the contrary. Thus, Bobbitt et al. does not disclose or suggest a master that is configured to, among other things, store an operation log that includes a record of changes to at least one of namespace data, which includes file identifiers for files for which the file data is stored as chunks, or mapping data, which maps the file identifiers to the chunks to which the file identifiers correspond, as recited in

claim 1.

Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, also do not disclose or suggest a master that is configured to, among other things, communicate with the servers at startup of the master to identify the chunks stored by the servers and record, in a non-persistent manner, information regarding the chunks stored by each of the servers as the location data, as further recited in claim 1.

The Examiner alleged that Dinker et al. discloses these features and cited column 6, lines 8-67, of Dinker et al. for support. Final Office Action, page 3. Appellants submit that the disclosure of Dinker et al. provides no support for the Examiner's allegation.

At column 6, lines 8-67, Dinker et al. discloses a replication topology manager that maintains the distribution of data on the nodes, as defined by a replication topology. Dinker et al. discloses that the replication topology manager can initiate one or more copy operations by the nodes so that the replication of data within the cluster conforms to the replication topology. Even assuming, for the sake of argument, that the nodes disclosed by Dinker et al. correspond to servers and that the replication topology manager corresponds to a master (points that Appellants do not concede), nowhere in this section, or elsewhere, does Dinker et al. disclose or suggest that the replication topology manager communicates with the nodes at startup to identify the data stored by the nodes. In fact, Dinker et al. does not specifically disclose that the replication topology manager performs any communication with the nodes to identify the data stored by the nodes, whether at startup or at another time. Thus, Dinker et al. does not disclose or suggest a master that is configured to, among other things, communicate with the servers at startup of the master to identify the chunks stored by the servers and record, in a non-persistent manner,

information regarding the chunks stored by each of the servers as the location data, as recited in claim 1.

Further, nowhere in the above-identified section, or elsewhere, does Dinker et al. disclose or suggest that the replication topology manager records, in a non-persistent manner, information regarding data stored by each of the nodes. Thus, Dinker et al. does not disclose or suggest a master that is configured to, among other things, communicate with the servers at startup of the master to identify the chunks stored by the servers and record, in a non-persistent manner, information regarding the chunks stored by each of the servers as the location data, as recited in claim 1.

The Examiner alleged that the communication interface, in Dinker et al., notifies the replication topology manager whenever changes in cluster membership are detected and, therefore, it is inherent that when a node, which is to become a manager, is added to the cluster, the node will have to receive the topology information. Final Office Action, page 17. Appellants submit that the Examiner's allegation lacks merit and falls short of establishing a rejection based on inherency.

According to M.P.E.P. § 2112, the fact that a certain result or characteristic may occur or be present in the prior art is not sufficient to establish the inherency of that result or characteristic. In relying upon the theory of inherency, the Examiner must provide a basis in fact and/or technical reasoning to reasonably support the determination that the allegedly inherent characteristic necessarily flows from the teachings of the applied prior art. Inherency cannot be established by probabilities or possibilities. The mere fact that a certain thing may result from a given set of circumstances is not sufficient. In this case, the Examiner's allegation does not meet

the requisite burden of proof to establish inherency. For example, the Examiner has provided absolutely no evidence that when a new node is added to a cluster in Dinker et al., that new node would necessarily communicate with the other nodes of the cluster, at startup of the new node, to identify the data stored by the nodes and record, in a non-persistent manner, information regarding the data stored by each of the nodes as the location data. Thus, contrary to the Examiner's allegation, Dinker et al. does not disclose or suggest a master that is configured to, among other things, communicate with the servers at startup of the master to identify the chunks stored by the servers and record, in a non-persistent manner, information regarding the chunks stored by each of the servers as the location data, as recited in claim 1.

The disclosure of Bobbitt et al. does not cure the deficiencies in the disclosure of Dinker et al. For example, Bobbitt et al. does not disclose or suggest a master that is configured to, among other things, communicate with the servers at startup of the master to identify the chunks stored by the servers and record, in a non-persistent manner, information regarding the chunks stored by each of the servers as the location data, as recited in claim 1. Bobbitt et al. actually teaches away from these features by disclosing the persistence of files when the computer turns on and off. Bobbitt et al., paragraph 0027.

In the Advisory Action, the Examiner repeated the above allegation based on inherency. Advisory Action, page 2. Appellants submit that the Examiner's allegation lacks merit for at least the reasons given above.

Thus, Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, do not disclose or suggest a master that is configured to, among other things, communicate with the servers at startup of the master to identify the chunks stored by the servers

and record, in a non-persistent manner, information regarding the chunks stored by each of the servers as the location data, as recited in claim 1.

The Examiner alleged that:

> It would have been obvious to one of ordinary skill in the art at the time of the invention to utilize the file system of Bobbitt to store the data chunks of Dinker. One would have been motivated do so in order to increase the efficiency of managing disk space by providing a manner in which additional servers can be added and data can be migrated (Bobbitt: see [0004]).

Final Office Action, page 4. Appellants submit that the Examiner's reason for combining Dinker et al. and Bobbitt et al. falls short of establishing a prima facie case of obviousness.

Appellants submit that the Examiner's allegation is merely a conclusory statement of an alleged benefit of the combination. Such conclusory statements have been repeatedly held to be insufficient for establishing a prima facie case of obviousness. In this respect, Appellants rely upon KSR International Co. v. Teleflex Inc., 550 U.S. 398 (April 30, 2007) (citing In re Kahn, 441 F.3d 977, 988 (Fed. Cir. 2006)), where it was held that rejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness. The Examiner's reason for combining Dinker et al. and Bobbitt et al. does not qualify as an articulated reason with some rational underpinning. Rather, the Examiner's reason is merely a conclusory statement.

Furthermore, the Examiner's reason for combining Dinker et al. and Bobbitt et al. lacks merit. Dinker et al. already discloses adding nodes (servers) and migrating data. Dinker et al., column 6, lines 8-23 and 39-42. Thus, the Examiner's reason for combining Dinker et al. and

Bobbitt et al. is not a valid reason for combining these references. Thus, the Examiner has not established a prima facie case of obviousness with regard to claim 1.

The multiple features expressly set forth in claim 1 allow for unique combinations of functionality and capability not present or anticipated by the Dinker et al. and Bobbitt et al. references, whether taken alone or in any reasonable combination.

For at least these reasons, it is respectfully submitted that claim 1 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 1 is respectfully requested.

Claims 3-10, 19, and 21 depend from claim 1. Claims 3-10, 19, and 21 are, therefore, patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103 for at least the reasons given with regard to claim 1. Reversal of the rejection of claims 3-10, 19, and 21 is respectfully requested.

2. Claim 11.

Dependent claim 11 recites that the master is configured to monitor the version numbers of the chunks stored by the servers.

Initially, claim 11 depends from claim 10. Therefore, claim 11 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, for at least the reasons given with regard to claim 10.

Further, Dinker et al. and Bobbitt et al. do not disclose or suggest the combination of features recited in claim 11. The Examiner alleged that Dinker et al. discloses the feature of claim 11 and cited column 6, lines 8-67, of Dinker et al. for support. Final Office Action, page

6. Appellants submit that the disclosure of Dinker et al. provides no support for the Examiner's allegation.

At column 6, lines 8-67, Dinker et al. discloses a replication topology manager that maintains the distribution of data on the nodes, as defined by a replication topology. Dinker et al. discloses that the replication topology manager can initiate one or more copy operations by the nodes so that the replication of data within the cluster conforms to the replication topology. Dinker et al. does not disclose that version numbers of the data are maintained. Thus, Dinker et al. does not disclose or suggest that the master is configured to monitor the version numbers of the chunks stored by the servers, as recited in claim 11.

In the Advisory Action, the Examiner alleged:

> Referring to Appellant's arguments on page 10 of the Remarks, the examiner respectfully disagrees. According to column 4, lines 47-59 of Dinker, the manager stores whether the nodes are primary or backup. A primary copy is considered to represent a first version and a secondary is considered to represent a second version.

Advisory Action, page 2. Appellants submit that the Examiner's allegation lacks merit. Claim 11 does not recite that multiple versions are stored. Rather, claim 11 recites that the master monitors information regarding the version numbers of chunks. Dinker et al. does not disclose or remotely suggest a master that monitors information regarding the version numbers of the chunks stored by the servers, as recited in claim 11.

For at least these reasons, it is respectfully submitted that claim 11 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 11 is respectfully requested.

3. Claim 13.

Independent claim 13 is directed to a master in a file system that includes the master connected to a plurality of servers. The master comprises means for communicating with the servers to identify file data stored by the servers as chunks; means for storing, in a non-persistent manner, location information that identifies ones of the servers that store the chunks; means for updating the location information by periodically instructing the servers to identify the data stored by the servers; means for storing namespace data that includes file identifiers for files for which the file data is stored as chunks by the servers; means for storing mapping data that maps the file identifiers to the chunks to which the file identifiers correspond; and means for storing an operation log that includes a record of changes to at least one of the namespace data or the mapping data.

Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, do not disclose or suggest the combination of features recited in claim 13. For example, Dinker et al. and Bobbitt et al. do not disclose or suggest means for storing, in a non-persistent manner, location information that identifies ones of the servers that store the chunks, as recited in claim 13.

The Examiner alleged that Dinker et al. discloses this feature and cited column 6, lines 8-67, of Dinker et al. for support. Final Office Action, page 6. Appellants submit that the disclosure of Dinker et al. provides no support for the Examiner's allegation.

At column 6, lines 8-67, Dinker et al. discloses a replication topology manager that maintains the distribution of data on the nodes, as defined by a replication topology. Dinker et al. discloses that the replication topology manager can initiate one or more copy operations by

the nodes so that the replication of data within the cluster conforms to the replication topology. Even assuming, for the sake of argument, that Dinker et al. discloses that the replication topology manager stores location information that identifies the data stored by the nodes (a point that Appellants do not concede), Dinker et al. does not disclose or remotely suggest that the location information is stored in a non-persistent manner. Thus, Dinker et al. cannot disclose or suggest means for storing, in a non-persistent manner, location information that identifies ones of the servers that store the chunks, as recited in claim 13.

The disclosure of Bobbitt et al. does not cure the deficiencies in the disclosure of Dinker et al. For example, Bobbitt et al. does not disclose or suggest means for storing, in a non-persistent manner, location information that identifies ones of the servers that store the chunks, as recited in claim 13. Bobbitt et al. actually teaches away from this feature by disclosing the persistence of files when the computer turns on and off. Bobbitt et al., paragraph 0027.

Dinker et al. and Bobbitt et al. also do not disclose or suggest means for storing an operation log that includes a record of changes to at least one of namespace data, which includes file identifiers for files for which the file data is stored as chunks by the servers, or mapping data, which maps the file identifiers to the chunks to which the file identifiers correspond, as further recited in claim 13.

The Examiner admitted that Dinker et al. does not disclose these features, and alleged that Bobbitt et al. discloses these features and cited paragraphs 0048 and 0052-0054 of Bobbitt et al. for support. Final Office Action, pages 6-7. Appellants submit that the disclosure of Bobbitt et al. provides no support for the Examiner's allegation for at least reasons similar to the reasons given with regard to claim 1.

The Examiner alleged that:

> It would have been obvious to one of ordinary skill in the art at the time of the invention to utilize the file system of Bobbitt to store the data chunks of Dinker. One would have been motivated do so in order to increase the efficiency of managing disk space by providing a manner in which additional servers can be added and data can be migrated (Bobbitt: see [0004]).

Final Office Action, page 7. Appellants submit that the Examiner's reason for combining Dinker et al. and Bobbitt et al. falls short of establishing a prima facie case of obviousness for at least reasons similar to the reasons given with regard to claim 1.

The multiple features expressly set forth in claim 13 allow for unique combinations of functionality and capability not present or anticipated by the Dinker et al. and Bobbitt et al. references, whether taken alone or in any reasonable combination.

For at least these reasons, it is respectfully submitted that claim 13 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 13 is respectfully requested.

4. Claim 15.

Independent claim 15 is directed to a file system that comprises a plurality of servers configured to store files as chunks; and a master connected to the servers and configured to store namespace data that includes file identifiers for which the files are stored as chunks, store mapping data that maps the file identifiers to the chunks to which the file identifiers correspond, store an operation log that includes a record of changes to at least one of the namespace data or the mapping data, and store location data that identifies which of the servers stores which of the chunks, where the master is configured to determine location information by communicating with the servers, the location information being based on which of the servers store ones of the

chunks, store the location information as the location data, and update the location data by periodically communicating with the servers to obtain changes to the location data.

Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, do not disclose or suggest the combination of features recited in claim 15. For example, Dinker et al. and Bobbitt et al. do not disclose or suggest a master that is configured to, among other things, store an operation log that includes a record of changes to at least one of the namespace data, which includes file identifiers for which the files are stored as chunks, or the mapping data, which maps the file identifiers to the chunks to which the file identifiers correspond, as recited in claim 15.

The Examiner admitted that Dinker et al. does not disclose or suggest an operation log, and alleged that Bobbitt et al. discloses storing an operation log that includes a record of changes to at least one of namespace data or mapping data, and cited paragraphs 0048 and 0052-0054 of Bobbitt et al. for support. Final Office Action, pages 7-8. Appellants submit that the disclosure of Bobbitt et al. provides no support for the Examiner's allegation for at least reasons similar to the reasons given with regard to claim 1.

Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, also do not disclose or suggest a master that is configured to, among other things, update the location data by periodically communicating with the servers to obtain changes to the location data, as further recited in claim 15.

The Examiner admitted that Dinker et al. does not disclose this feature, and alleged that Bobbitt et al. discloses this feature and cited paragraph 0080 of Bobbitt et al. for support. Final Office Action, page 7-8. Appellants submit that the disclosure of Bobbitt et al. provides no

support for the Examiner's allegation.

At paragraph 0080, Bobbitt et al. discloses:

> Further details of Gossamer client agent 43 are shown in FIG. 8. As indicated by an agent module 82, Gossamer client agent 43 functions as a UNIX agent that performs polling for configuration changes, mounts gtrees (i.e., mounts the underlying file system corresponding to the gtree), and provides an interface for a centralized administration module to communicate with the GVFD module. Agent module 82 communicates with GVFD 42 via a driver communication module 84, which provides a driver communication interface, and enables GVVs and gtrees to be added and removed. Agent module 82 is also enabled to access configuration information 45.

In this section, Bobbitt et al. discloses a client agent, on a client device, that performs polling for configuration changes. This disclosure of Bobbitt et al. is deficient for at least a couple of reasons. First, the Examiner alleged that the master file server, disclosed by Bobbitt et al. at paragraph 0084, corresponds to the master recited in claim 15. Final Office Action, page 8. The client agent that performs polling is located on a client device. Bobbitt et al., Figure 8. Bobbitt et al. does not disclose or suggest that the client agent is located on the master file server. Thus, Bobbitt et al. does not disclose or suggest a master that is configured to, among other things, update the location data by periodically communicating with the servers to obtain changes to the location data, as further recited in claim 15.

Also, Bobbitt et al. does not disclose or remotely suggest that polling for configuration changes includes periodically communicating with servers to obtain changes to location data, which reflects which of the servers store ones of the chunks. Any argument to the contrary is based solely on impermissible hindsight. Thus, even if the disclosure of Bobbitt et al. could reasonably be construed as disclosing that the client agent is located on the master file server (a point that Appellants do not concede), Bobbitt et al. still would not disclose that the client agent

periodically communicates with servers to obtain changes to location data, which reflects which of the servers store ones of the chunks. Thus, Bobbitt et al. does not disclose or suggest a master that is configured to, among other things, update the location data by periodically communicating with the servers to obtain changes to the location data, as further recited in claim 15.

The Examiner alleged that:

> It would have been obvious to one of ordinary skill in the art at the time of the invention to utilize the file system of Bobbitt to store the data chunks of Dinker. One would have been motivated do so in order to increase the efficiency of managing disk space by providing a manner in which additional servers can be added and data can be migrated (Bobbitt: see [0004]).

Final Office Action, page 8. Appellants submit that the Examiner's reason for combining Dinker et al. and Bobbitt et al. falls short of establishing a prima facie case of obviousness for at least reasons similar to the reasons given with regard to claim 1.

The multiple features expressly set forth in claim 15 allow for unique combinations of functionality and capability not present or anticipated by the Dinker et al. and Bobbitt et al. references, whether taken alone or in any reasonable combination.

For at least these reasons, it is respectfully submitted that claim 15 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 15 is respectfully requested.

Claim 34 depends from claim 15. Claim 34 is, therefore, patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103 for at least the reasons given with regard to claim 15. Reversal of the rejection of claim 34 is respectfully requested.

5. Claim 16.

Independent claim 16 is directed to a file system that comprises a plurality of servers configured to store file data as chunks; and a master connected to the servers and configured to store namespace data that includes file identifiers for files for which the file data is stored as chunks, store mapping data that maps the file identifiers to the chunks to which the file identifiers correspond, store an operation log that includes a record of changes to the namespace data and the mapping data, and store location data that identifies which of the servers stores which of the chunks, where the master is configured to communicate with the servers to determine location information of the data, the location information being based on which of the servers store the chunks, and store the location information as the location data.

Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, do not disclose or suggest the combination of features recited in claim 16. For example, Dinker et al. and Bobbitt et al. do not disclose or suggest a master that is configured to, among other things, store an operation log that includes a record of changes to the namespace data, which includes file identifiers for files for which the file data is stored as chunks, and the mapping data, which maps the file identifiers to the chunks to which the file identifiers correspond, as recited in claim 16.

The Examiner admitted that Dinker et al. does not disclose or suggest an operation log, and alleged that Bobbitt et al. discloses storing an operation log that includes a record of changes to at least one of namespace data or mapping data, and cited paragraphs 0048 and 0052-0054 of Bobbitt et al. for support. Final Office Action, page 9. Appellants note that claim 16 does not recite an operation log that includes a record of changes to at least one of namespace data or

mapping data, as alleged by the Examiner. Rather, claim 16 recites an operation log that includes a record of changes to namespace data AND mapping data. Bobbitt et al. does not disclose or suggest this feature for at least reasons similar to the reasons given with regard to claim 1.

Further, even if the disclosure of Bobbitt et al. could somehow be construed as disclosing storing an operation log that includes a record of changes to the namespace data (a point that Appellants do not concede), as alleged by the Examiner, Bobbitt et al. does not disclose or remotely suggest storing an operation log that includes a record of changes to the namespace data and the mapping data, as recited in claim 16. The Examiner did not address this feature and, thus, did not establish a prima facie case of obviousness with regard to claim 16.

The Examiner alleged that:

> It would have been obvious to one of ordinary skill in the art at the time of the invention to utilize the file system of Bobbitt to store the data chunks of Dinker. One would have been motivated do so in order to increase the efficiency of managing disk space by providing a manner in which additional servers can be added and data can be migrated (Bobbitt: see [0004]).

Final Office Action, page 10. Appellants submit that the Examiner's reason for combining Dinker et al. and Bobbitt et al. falls short of establishing a prima facie case of obviousness for at least reasons similar to the reasons given with regard to claim 1.

The multiple features expressly set forth in claim 16 allow for unique combinations of functionality and capability not present or anticipated by the Dinker et al. and Bobbitt et al. references, whether taken alone or in any reasonable combination.

For at least these reasons, it is respectfully submitted that claim 16 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 16 is respectfully requested.

Claim 36 depends from claim 16. Claim 36 is, therefore, patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103 for at least the reasons given with regard to claim 16. Reversal of the rejection of claim 36 is respectfully requested.

6. Claim 20.

Dependent claim 20 recites that the master stores the namespace data using prefix-compression.

Initially, claim 20 depends from claim 1. Therefore, claim 20 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, for at least the reasons given with regard to claim 1.

Further, Dinker et al. and Bobbitt et al. do not disclose or suggest the combination of features recited in claim 20. The Examiner alleged that Bobbitt et al. discloses the feature of claim 20 and cited paragraphs 0052-0054, of Bobbitt et al. for support. Final Office Action, page 10. Appellants submit that the disclosure of Bobbitt et al. provides no support for the Examiner's allegation.

At paragraphs 0052-0054, Bobbitt et al. discloses that translations between virtual pathname and actual server-pathname are handled through a GVV master directory structure that logically divides its data into three spaces: a Gossamer namespace, a temporary migrating space, and a garbage space. Bobbitt et al. discloses that the directory structure stored in the Gossamer

namespace parallels the virtual directory hierarchy, where the files contained in the virtual directories are replaced by file pointers having the same names as the original files. Bobbitt et al. discloses that each of the file pointers comprises two pieces of information: a file GUID and a GUID slave location identifier. Nowhere in this section, or elsewhere, does Bobbitt et al. disclose or remotely suggest using a prefix-compression technique, let alone storing namespace data using prefix-compression. Thus, Bobbitt et al. does not disclose or suggest a master that stores the namespace data using prefix-compression, as recited in claim 20.

Further, Appellants submit that the Examiner has not established a prima facie case of obviousness with regard to claim 20. For example, the Examiner has not provided even a single reason why one of ordinary skill in the art, at the time of Appellants' invention, would have modified the Dinker et al. system to store namespace data using prefix-compression, as allegedly disclosed by Bobbitt et al. Thus, the Examiner has not established a prima facie case of obviousness with regard to claim 20.

For at least these reasons, it is respectfully submitted that claim 20 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 20 is respectfully requested.

7. Claim 22.

Dependent claim 22 recites that the chunk handle, which uniquely identifies one of the chunks, encodes a timestamp.

Initially, claim 22 depends from claim 21. Therefore, claim 22 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, for at least the reasons given with regard to claim 21.

Further, Dinker et al. and Bobbitt et al. do not disclose or suggest the combination of features recited in claim 22. The Examiner alleged that Bobbitt et al. discloses this feature and cited paragraph 0054 of Bobbitt et al. for support. Final Office Action, page 10. Appellants submit that the disclosure of Bobbitt et al. provides no support for the Examiner's allegation.

At paragraph 0054, Bobbitt et al. discloses a globally unique identifier (GUID) assigned to a file. Nowhere does Bobbitt et al. disclose or remotely suggest that the GUID encodes a timestamp. Rather Bobbitt et al. discloses that the GUID is merely a 128-bit identifier. Bobbitt et al., paragraph 0055. Thus, Bobbitt et al. does not disclose or suggest a chunk handle that encodes a timestamp, as recited in claim 22.

In the Advisory Action, the Examiner alleged that "the GUID is considered to include the timestamp." Advisory Action, page 2. Appellants submit that there is no merit to the Examiner's allegation. The Examiner has provided no evidence that the GUID, disclosed by Bobbitt et al., includes anything more than the 128-bit identifier that Bobbitt et al. discloses that the GUID includes.

Further, Appellants submit that the Examiner has not established a prima facie case of obviousness with regard to claim 22. For example, the Examiner has not provided even a single reason why one of ordinary skill in the art, at the time of Appellants' invention, would have modified the Dinker et al. system to include a GUID that encodes a timestamp, as allegedly disclosed by Bobbitt et al. Thus, the Examiner has not established a prima facie case of obviousness with regard to claim 22.

For at least these reasons, it is respectfully submitted that claim 22 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 22 is respectfully requested.

8. Claim 23.

Dependent claim 23 recites that the master is configured to update the location data by periodically instructing the servers to provide information regarding the chunks stored by the servers.

Initially, claim 23 depends from claim 1. Therefore, claim 23 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, for at least the reasons given with regard to claim 1.

Further, Dinker et al. and Bobbitt et al. do not disclose or suggest the combination of features recited in claim 23. The Examiner alleged that Dinker et al. discloses this feature and cited column 6, lines 8-67, of Dinker et al. for support. Final Office Action, page 10. Appellants submit that the disclosure of Dinker et al. provides no support for the Examiner's allegation.

Initially, Appellants note that the Examiner admitted, with regard to claim 15, that Dinker et al. does not disclose or suggest this feature. Final Office Action, pages 7-8. With regard to claim 23, however, the Examiner alleged that Dinker et al. does in fact disclose this feature. Final Office Action, page 10. Appellants object to the Examiner's inconsistent statements provided in the final Office Action.

At column 6, lines 8-67, Dinker et al. discloses a replication topology manager that maintains the distribution of data on the nodes, as defined by a replication topology. Dinker et al. discloses that the replication topology manager can initiate one or more copy operations by

the nodes so that the replication of data within the cluster conforms to the replication topology. Even assuming, for the sake of argument, that the nodes disclosed by Dinker et al. correspond to servers and that the replication topology manager corresponds to a master (points that Appellants do not concede), nowhere in this section, or elsewhere, does Dinker et al. disclose or suggest that the replication topology manager periodically instructs the nodes to provide information regarding the data stored by the nodes, as would be required based on the Examiner's apparent interpretation of Dinker et al. In fact, Dinker et al. does not specifically disclose that the replication topology manager performs any communication with the nodes to identify the data stored by the nodes, whether at startup or at another time. Thus, Dinker et al. does not disclose or suggest a master that is configured to update the location data by periodically instructing the servers to provide information regarding the chunks stored by the servers, as recited in claim 23.

For at least these reasons, it is respectfully submitted that claim 23 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 23 is respectfully requested.

9. Claim 24.

Dependent claim 24 recites that the operation log includes a logical timeline that defines an order for concurrent operations.

Initially, claim 24 depends from claim 1. Therefore, claim 24 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, for at least the reasons given with regard to claim 1.

Further, Dinker et al. and Bobbitt et al. do not disclose or suggest the combination of features recited in claim 24. The Examiner alleged that Bobbitt et al. discloses this feature and

cited paragraph 0048 of Bobbitt et al. for support. Final Office Action, page 11. Appellants submit that the disclosure of Bobbitt et al. provides no support for the Examiner's allegation.

At paragraph 0048, Bobbitt et al. discloses configuration information that identifies what server hosts various gtrees, what devices store the master and slave gtrees, the exports provided by each server, the roles the various components play, schedule data, status files pertaining to operations in progress, and log files. As explained above with regard to claim 1, while this section of Bobbitt et al. mentions "log files," nowhere does Bobbitt et al. disclose or remotely suggest that these log files include a record of changes to at least one of namespace data, which includes file identifiers for files for which the file data is stored as chunks, or mapping data, which maps the file identifiers to the chunks to which the file identifiers correspond. Thus, Bobbitt et al. cannot disclose or suggest an operation log that includes a logical timeline that defines an order for concurrent operations, as recited in claim 24.

Even assuming, for the sake of argument, that the disclosure of Bobbitt et al. could somehow be construed as disclosing an operation log includes a record of changes to at least one of namespace data or mapping data (a point that Appellants do not concede for at least the reasons given above), Bobbitt et al.'s mere disclosure of "log files" falls short of establishing that these "log files" include a logical timeline that defines an order for concurrent operations. Thus, Bobbitt et al. does not disclose or suggest an operation log that includes a logical timeline that defines an order for concurrent operations, as recited in claim 24.

In the Advisory Action, the Examiner indicated that for further clarification, "see Table 1 of Bobbit." Advisory Action, page 2. Table 1 of Bobbitt et al. merely provides a list of types of VFS/vnode requests. Nothing in Table 1 of Bobbitt et al. reasonably corresponds to an operation

log, let alone an operation log that includes a logical timeline that defines an order for concurrent operations, as recited in claim 24.

Further, Appellants submit that the Examiner has not established a prima facie case of obviousness with regard to claim 24. For example, the Examiner has not provided even a single reason why one of ordinary skill in the art, at the time of Appellants' invention, would have modified the Dinker et al. system to include an operation log that includes a logical timeline that defines an order for concurrent operations, as allegedly disclosed by Bobbitt et al. Thus, the Examiner has not established a prima facie case of obviousness with regard to claim 24.

For at least these reasons, it is respectfully submitted that claim 24 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 24 is respectfully requested.

10. Claim 27.

Dependent claim 27 recites that the operation log includes a logical timeline that defines an order for concurrent operations.

Initially, claim 27 depends from claim 13. Therefore, claim 27 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, for at least the reasons given with regard to claim 13.

Further, Dinker et al. and Bobbitt et al. do not disclose or suggest the combination of features recited in claim 27. The Examiner alleged that Bobbitt et al. discloses this feature and cited paragraph 0048 of Bobbitt et al. for support. Final Office Action, page 11. Appellants submit that the disclosure of Bobbitt et al. provides no support for the Examiner's allegation for at least reasons similar to the reasons given with regard to claim 24.

Further, Appellants submit that the Examiner has not established a prima facie case of obviousness with regard to claim 27. For example, the Examiner has not provided even a single reason why one of ordinary skill in the art, at the time of Appellants' invention, would have modified the Dinker et al. system to include an operation log that includes a logical timeline that defines an order for concurrent operations, as allegedly disclosed by Bobbitt et al. Thus, the Examiner has not established a prima facie case of obviousness with regard to claim 27.

For at least these reasons, it is respectfully submitted that claim 27 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 27 is respectfully requested.

11. Claim 30.

Independent claim 30 is directed to a method performed by a master device in a file system that includes the master device connected to a plurality of server devices. The method comprises communicating with the server devices to identify file data stored by the server devices as chunks; storing location information that identifies ones of the server devices that store the chunks; storing namespace data that includes file identifiers for files for which the file data is stored as chunks by the server devices; storing mapping data that maps the file identifiers to the chunks to which the file identifiers correspond; and maintaining an operation log that includes a record of changes to the namespace data and the mapping data.

Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, do not disclose or suggest the combination of features recited in claim 30. For example, Dinker et al. and Bobbitt et al. do not disclose or suggest maintaining an operation log that includes a record of changes to the namespace data, which includes file identifiers for files for which the

file data is stored as chunks by the server devices, and the mapping data, which maps the file identifiers to the chunks to which the file identifiers correspond, as recited in claim 30.

The Examiner admitted that Dinker et al. does not disclose or suggest an operation log, and alleged that Bobbitt et al. discloses storing an operation log that includes a record of changes to at least one of namespace data or mapping data, and cited paragraphs 0048 and 0052-0054 of Bobbitt et al. for support. Final Office Action, pages 11-12. Appellants note that claim 30 does not recite an operation log that includes a record of changes to at least one of namespace data or mapping data, as alleged by the Examiner. Rather, claim 30 recites an operation log that includes a record of changes to namespace data AND mapping data. Bobbitt et al. does not disclose or suggest this feature for at least reasons similar to the reasons given with regard to claim 1.

Further, even if the disclosure of Bobbitt et al. could somehow be construed as disclosing storing an operation log that includes a record of changes to the namespace data (a point that Appellants do not concede), as alleged by the Examiner, Bobbitt et al. does not disclose or remotely suggest storing an operation log that includes a record of changes to the namespace data and the mapping data, as recited in claim 30. The Examiner did not address this feature and, thus, did not establish a prima facie case of obviousness with regard to claim 30.

The Examiner alleged that:

> It would have been obvious to one of ordinary skill in the art at the time of the invention to utilize the file system of Bobbitt to store the data chunks of Dinker. One would have been motivated do so in order to increase the efficiency of managing disk space by providing a manner in which additional servers can be added and data can be migrated (Bobbitt: see [0004]).

Final Office Action, page 12. Appellants submit that the Examiner's reason for combining

Dinker et al. and Bobbitt et al. falls short of establishing a prima facie case of obviousness for at least reasons similar to the reasons given with regard to claim 1.

The multiple features expressly set forth in claim 30 allow for unique combinations of functionality and capability not present or anticipated by the Dinker et al. and Bobbitt et al. references, whether taken alone or in any reasonable combination.

For at least these reasons, it is respectfully submitted that claim 30 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 30 is respectfully requested.

12. Claim 31.

Dependent claim 31 recites that maintaining the operation log includes storing a logical timeline that defines an order for operations including concurrent operations.

Initially, claim 31 depends from claim 30. Therefore, claim 31 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, for at least the reasons given with regard to claim 30.

Further, Dinker et al. and Bobbitt et al. do not disclose or suggest the combination of features recited in claim 31. The Examiner alleged that Bobbitt et al. discloses this feature and cited paragraph 0048 of Bobbitt et al. for support. Final Office Action, page 12. Appellants submit that the disclosure of Bobbitt et al. provides no support for the Examiner's allegation for at least reasons similar to the reasons given with regard to claim 24.

Further, Appellants submit that the Examiner has not established a prima facie case of obviousness with regard to claim 31. For example, the Examiner has not provided even a single reason why one of ordinary skill in the art, at the time of Appellants' invention, would have

modified the Dinker et al. system to include an operation log that includes a logical timeline that defines an order for concurrent operations, as allegedly disclosed by Bobbitt et al. Thus, the Examiner has not established a prima facie case of obviousness with regard to claim 31.

For at least these reasons, it is respectfully submitted that claim 31 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 31 is respectfully requested.

13. Claim 35.

Dependent claim 35 recites that the master is further configured to identify one or more of the servers to store a new chunk based on failure correlation properties associated with the servers, and place the new chunk at the identified one or more servers.

Initially, claim 35 depends from claim 15. Therefore, claim 35 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, for at least the reasons given with regard to claim 15.

Further, Dinker et al. and Bobbitt et al. do not disclose or suggest the combination of features recited in claim 35. The Examiner alleged that Bobbitt et al. discloses these features and cited column 6, lines 8-67, and column 6, line 27 – column 7, line 3, of Dinker et al., and paragraph 0081, lines 10-14, of Bobbitt et al. for support. Final Office Action, pages 12-13. Appellants submit that the disclosures of Dinker et al. and Bobbitt et al. provide no support for the Examiner's allegations.

At column 6, line 8 – column 7, line 3, Dinker et al. discloses a replication topology manager that maintains the distribution of data on the nodes, as defined by a replication topology. Dinker et al. discloses that the replication topology manager can initiate one or more

copy operations by the nodes so that the replication of data within the cluster conforms to the replication topology. Nowhere in this section, or elsewhere, does Dinker et al. disclose or suggest that the replication topology manager performs one of these copy operations based on the failure correlation properties associated with the nodes. Rather, Dinker et al. merely discloses copying data from a failed node to another node. Dinker et al., column 6, lines 44-49. Thus, Dinker et al. does not disclose or suggest a master that is further configured to identify one or more of the servers to store a new chunk based on failure correlation properties associated with the servers, and place the new chunk at the identified one or more servers, as recited in claim 35.

At paragraph 0081, lines 10-14, Bobbitt et al. discloses:

> In general, the slave for storing a new file may be selected using various criteria, including storage space and load-balancing considerations. In one embodiment, new files are stored on the slave with the largest free disk space.

In this section, Bobbitt et al. discloses selecting a slave to store a new file based on storage space and load-balancing considerations. Nowhere in this section, or elsewhere, does Bobbitt et al. disclose taking failure correlation properties into consideration when selecting a server to store a new file. The amount of available storage space available at the servers and load-balancing considerations (i.e., the amount of load on the servers) do not reasonably correspond to failure correlation properties. Appellants' specification describes failure correlation properties as system conditions that may concurrently affect the availability of two or more chunk servers, such as placement of the chunk servers on the same or different racks. Appellants' specification, pages 15-16. Thus, Bobbitt et al. does not disclose or suggest a master that is further configured to identify one or more of the servers to store a new chunk based on failure correlation properties

associated with the servers, and place the new chunk at the identified one or more servers, as recited in claim 35.

Further, Appellants submit that the Examiner has not established a prima facie case of obviousness with regard to claim 35. For example, the Examiner has not provided even a single reason why one of ordinary skill in the art, at the time of Appellants' invention, would have modified the Dinker et al. system to include a master that is further configured to identify one or more of the servers to store a new chunk based on failure correlation properties associated with the servers, and place the new chunk at the identified one or more servers, as allegedly disclosed by Bobbitt et al. Thus, the Examiner has not established a prima facie case of obviousness with regard to claim 35.

For at least these reasons, it is respectfully submitted that claim 35 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 35 is respectfully requested.

14. Claim 37.

Dependent claim 37 recites that the master is further configured to identify one or more of the servers to store a new chunk based on failure correlation properties associated with the servers, and place the new chunk at the identified one or more servers.

Initially, claim 37 depends from claim 16. Therefore, claim 37 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, for at least the reasons given with regard to claim 16.

Further, Dinker et al. and Bobbitt et al. do not disclose or suggest the combination of features recited in claim 37. The Examiner alleged that Bobbitt et al. discloses these features and

cited column 6, line 8 – column 7, line 3, of Dinker et al., and paragraph 0081, lines 10-14, of Bobbitt et al. for support. Final Office Action, pages 12-13. Appellants submit that the disclosures of Dinker et al. and Bobbitt et al. provide no support for the Examiner's allegations for at least reasons similar to the reasons given with regard to claim 35.

Further, Appellants submit that the Examiner has not established a prima facie case of obviousness with regard to claim 37. For example, the Examiner has not provided even a single reason why one of ordinary skill in the art, at the time of Appellants' invention, would have modified the Dinker et al. system to include a master that is further configured to identify one or more of the servers to store a new chunk based on failure correlation properties associated with the servers, and place the new chunk at the identified one or more servers, as allegedly disclosed by Bobbitt et al. Thus, the Examiner has not established a prima facie case of obviousness with regard to claim 37.

For at least these reasons, it is respectfully submitted that claim 37 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 37 is respectfully requested.

15. Claim 38.

Dependent claim 38 recites means for identifying one or more of the servers to store a new chunk based on utilization of the servers, prior chunk distribution involving the servers, and failure correlation properties associated with the servers; and means for placing the new chunk at the identified one or more servers.

Initially, claim 38 depends from claim 13. Therefore, claim 38 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, for at least the reasons given with regard to claim 13.

Further, Dinker et al. and Bobbitt et al. do not disclose or suggest the combination of features recited in claim 38. The Examiner alleged that Bobbitt et al. discloses these features and cited column 6, line 8 – column 7, line 3, column 5, lines 5-6, and column 10, line 67 – column 11, line 3, of Dinker et al., and paragraph 0081, lines 10-14, of Bobbitt et al. for support. Final Office Action, page 13. Appellants submit that the disclosures of Dinker et al. and Bobbitt et al. provide no support for the Examiner's allegations.

At column 6, line 8 – column 7, line 3, Dinker et al. discloses a replication topology manager that maintains the distribution of data on the nodes, as defined by a replication topology. Dinker et al. discloses that the replication topology manager can initiate one or more copy operations by the nodes so that the replication of data within the cluster conforms to the replication topology. Nowhere in this section, or elsewhere, does Dinker et al. disclose or suggest that the replication topology manager performs one of these copy operations based on (1) utilization of the nodes, (2) prior chunk distribution involving the nodes, and (3) failure correlation properties associated with the nodes. Rather, Dinker et al. merely discloses copying data from a failed node to another node. Dinker et al., column 6, lines 44-49. Thus, Dinker et al. does not disclose or suggest means for identifying one or more of the servers to store a new chunk based on utilization of the servers, prior chunk distribution involving the servers, and failure correlation properties associated with the servers; and means for placing the new chunk at the identified one or more servers, as recited in claim 38.

At column 5, lines 5-6, Dinker et al. discloses that new data may be sent to the node currently storing the least amount of data. This disclosure of Dinker et al. can, at best, correspond to utilization of the node. Nowhere in this section, or elsewhere, does Dinker et al. disclose or suggest that new data is sent to a node based on (1) utilization of the node, (2) prior chunk distribution involving the node, and (3) failure correlation properties associated with the node. Thus, Dinker et al. does not disclose or suggest means for identifying one or more of the servers to store a new chunk based on utilization of the servers, prior chunk distribution involving the servers, and failure correlation properties associated with the servers; and means for placing the new chunk at the identified one or more servers, as recited in claim 38.

At column 10, line 67 – column 11, line 3, Dinker et al. discloses that the replication topology may provide load balancing by distributing data so that each node handles a similar volume of client access requests during a given time. This disclosure of Dinker et al. can, at best, correspond to utilization of the node. Nowhere in this section, or elsewhere, does Dinker et al. disclose or suggest that new data is sent to a node based on (1) utilization of the node, (2) prior chunk distribution involving the node, and (3) failure correlation properties associated with the node. Thus, Dinker et al. does not disclose or suggest means for identifying one or more of the servers to store a new chunk based on utilization of the servers, prior chunk distribution involving the servers, and failure correlation properties associated with the servers; and means for placing the new chunk at the identified one or more servers, as recited in claim 38.

At paragraph 0081, lines 10-14, Bobbitt et al. discloses:

> In general, the slave for storing a new file may be selected using various criteria, including storage space and load-balancing considerations. In one embodiment, new files are stored on the slave with the largest free disk space.

In this section, Bobbitt et al. discloses selecting a slave to store a new file based on storage space and load-balancing considerations. This disclosure of Bobbitt et al., at best, corresponds to utilization of the slaves. Nowhere in this section, or elsewhere, does Bobbitt et al. disclose or suggest selecting a slave to store a new file based on (1) utilization of the slave, (2) prior chunk distribution involving the slave, and (3) failure correlation properties associated with the slave. Thus, Bobbitt et al. does not disclose or suggest means for identifying one or more of the servers to store a new chunk based on utilization of the servers, prior chunk distribution involving the servers, and failure correlation properties associated with the servers; and means for placing the new chunk at the identified one or more servers, as recited in claim 38.

Further, Appellants submit that the Examiner has not established a prima facie case of obviousness with regard to claim 38. For example, the Examiner has not provided even a single reason why one of ordinary skill in the art, at the time of Appellants' invention, would have modified the Dinker et al. system to include means for identifying one or more of the servers to store a new chunk based on utilization of the servers, prior chunk distribution involving the servers, and failure correlation properties associated with the servers; and means for placing the new chunk at the identified one or more servers, as allegedly disclosed by Bobbitt et al. Thus, the Examiner has not established a prima facie case of obviousness with regard to claim 38.

For at least these reasons, it is respectfully submitted that claim 38 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 38 is respectfully requested.

16. Claim 39.

Dependent claim 39 recites identifying one or more of the server devices to store a new chunk based on utilization of the server devices, prior chunk distribution involving the server devices, and failure correlation properties associated with the server devices; and placing the new chunk at the identified one or more server devices.

Initially, claim 39 depends from claim 30. Therefore, claim 39 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, for at least the reasons given with regard to claim 30.

Further, Dinker et al. and Bobbitt et al. do not disclose or suggest the combination of features recited in claim 39. The Examiner alleged that Bobbitt et al. discloses these features and cited column 6, line 8 – column 7, line 3, column 5, lines 5-6, and column 10, line 67 – column 11, line 3, of Dinker et al., and paragraph 0081, lines 10-14, of Bobbitt et al. for support. Final Office Action, page 14. Appellants submit that the disclosures of Dinker et al. and Bobbitt et al. provide no support for the Examiner's allegations for at least reasons similar to the reasons given with regard to claim 38.

Further, Appellants submit that the Examiner has not established a prima facie case of obviousness with regard to claim 39. For example, the Examiner has not provided even a single reason why one of ordinary skill in the art, at the time of Appellants' invention, would have modified the Dinker et al. system to include identifying one or more of the server devices to store a new chunk based on utilization of the server devices, prior chunk distribution involving the server devices, and failure correlation properties associated with the server devices; and placing the new chunk at the identified one or more server devices, as allegedly disclosed by

Bobbitt et al. Thus, the Examiner has not established a prima facie case of obviousness with regard to claim 39.

For at least these reasons, it is respectfully submitted that claim 39 is patentable over Dinker et al. and Bobbitt et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 39 is respectfully requested.

B. The Rejection Under 35 U.S.C. § 103(a) Based on Dinker et al. in View of Bobbitt et al. and Rao et al. Should be Reversed.

1. Claim 25.

Dependent claim 25 recites that the master is configured to determine when a size of the operation log exceeds a threshold, and create a checkpoint of the operation log when the size of the operation log exceeds the threshold.

Initially, claim 25 depends from claim 1. The disclosure of Rao et al. does not cure the deficiencies in the disclosures of Dinker et al. and Bobbitt et al. identified above with regard to claim 1. Therefore, claim 25 is patentable over Dinker et al., Bobbitt et al., and Rao et al., whether taken alone or in any reasonable combination, for at least the reasons given with regard to claim 1.

Further, the Examiner alleged that it would have been obvious to include the alleged disclosure of Rao et al. in the combined system of Dinker et al. and Bobbitt et al. "in order to increase the efficiency of the system." Final Office Action, page 15. Appellants submit that the Examiner's reason for combining Dinker et al., Bobbitt et al., and Rao et al. falls short of establishing a prima facie case of obviousness.

Appellants submit that the Examiner's allegation is merely a conclusory statement of an alleged benefit of the combination. Such conclusory statements have been repeatedly held to be insufficient for establishing a prima facie case of obviousness. In this respect, Appellants rely upon KSR International Co. v. Teleflex Inc., 550 U.S. 398 (April 30, 2007) (citing In re Kahn, 441 F.3d 977, 988 (Fed. Cir. 2006)), where it was held that rejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness. The Examiner's reason for combining Dinker et al., Bobbitt et al., and Rao et al. does not qualify as an articulated reason with some rational underpinning. Rather, the Examiner's reason is merely a conclusory statement.

Furthermore, the Examiner did not explain how determining when a size of an operation log exceeds a threshold and creating a checkpoint of the operation log when the size of the operation log exceeds the threshold, as allegedly disclosed by Rao et al., would increase the efficiency of the combined Dinker et al. and Bobbitt et al. system. Thus, the Examiner has not established a prima facie case of obviousness with regard to claim 25.

For at least these reasons, it is respectfully submitted that claim 25 is patentable over Dinker et al., Bobbitt et al., and Rao et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 25 is respectfully requested.

2. Claim 26.

Dependent claim 26 recites that the master is configured to create a new operation log file, and create the checkpoint as a background operation.

Initially, claim 26 depends from claim 25. The disclosure of Rao et al. does not cure the deficiencies in the disclosures of Dinker et al. and Bobbitt et al. identified above with regard to claim 25. Therefore, claim 26 is patentable over Dinker et al., Bobbitt et al., and Rao et al., whether taken alone or in any reasonable combination, for at least the reasons given with regard to claim 25.

Further, Appellants submit that the Examiner has not established a prima facie case of obviousness with regard to claim 26. For example, the Examiner has not provided even a single reason why one of ordinary skill in the art, at the time of Appellants' invention, would have modified the Dinker et al. system to create a new operation log file and create a checkpoint as a background operation, as allegedly disclosed by Rao et al. Thus, the Examiner has not established a prima facie case of obviousness with regard to claim 26.

For at least these reasons, it is respectfully submitted that claim 26 is patentable over Dinker et al., Bobbitt et al., and Rao et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 26 is respectfully requested.

3. Claim 28.

Dependent claim 28 recites means for determining when a size of the operation log exceeds a threshold, and means for creating a checkpoint of the operation log when the size of the operation log exceeds the threshold.

Initially, claim 28 depends from claim 13. The disclosure of Rao et al. does not cure the deficiencies in the disclosures of Dinker et al. and Bobbitt et al. identified above with regard to claim 13. Therefore, claim 28 is patentable over Dinker et al., Bobbitt et al., and Rao et al.,

whether taken alone or in any reasonable combination, for at least the reasons given with regard to claim 13.

Further, the Examiner alleged that it would have been obvious to include the alleged disclosure of Rao et al. in the combined system of Dinker et al. and Bobbitt et al. "in order to increase the efficiency of the system." Final Office Action, page 15. Appellants submit that the Examiner's reason for combining Dinker et al., Bobbitt et al., and Rao et al. falls short of establishing a prima facie case of obviousness for at least reasons similar to the reasons given with regard to claim 25.

For at least these reasons, it is respectfully submitted that claim 28 is patentable over Dinker et al., Bobbitt et al., and Rao et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 28 is respectfully requested.

4. Claim 29.

Dependent claim 29 recites that the means for creating the checkpoint includes means for creating a new operation log file, and means for creating the checkpoint as a background operation.

Initially, claim 29 depends from claim 28. The disclosure of Rao et al. does not cure the deficiencies in the disclosures of Dinker et al. and Bobbitt et al. identified above with regard to claim 28. Therefore, claim 29 is patentable over Dinker et al., Bobbitt et al., and Rao et al., whether taken alone or in any reasonable combination, for at least the reasons given with regard to claim 28.

Further, Appellants submit that the Examiner has not established a prima facie case of obviousness with regard to claim 29. For example, the Examiner has not provided even a single reason why one of ordinary skill in the art, at the time of Appellants' invention, would have modified the Dinker et al. system to create a new operation log file and create a checkpoint as a background operation, as allegedly disclosed by Rao et al. Thus, the Examiner has not established a prima facie case of obviousness with regard to claim 29.

For at least these reasons, it is respectfully submitted that claim 29 is patentable over Dinker et al., Bobbitt et al., and Rao et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 29 is respectfully requested.

5. Claim 32.

Dependent claim 32 recites determining when a size of the operation log exceeds a threshold, and creating a checkpoint of the operation log when the size of the operation log exceeds the threshold.

Initially, claim 32 depends from claim 30. The disclosure of Rao et al. does not cure the deficiencies in the disclosures of Dinker et al. and Bobbitt et al. identified above with regard to claim 30. Therefore, claim 32 is patentable over Dinker et al., Bobbitt et al., and Rao et al., whether taken alone or in any reasonable combination, for at least the reasons given with regard to claim 30.

Further, the Examiner alleged that it would have been obvious to include the alleged disclosure of Rao et al. in the combined system of Dinker et al. and Bobbitt et al. "in order to increase the efficiency of the system." Final Office Action, page 15. Appellants submit that the

Examiner's reason for combining Dinker et al., Bobbitt et al., and Rao et al. falls short of establishing a prima facie case of obviousness for at least reasons similar to the reasons given with regard to claim 25.

For at least these reasons, it is respectfully submitted that claim 32 is patentable over Dinker et al., Bobbitt et al., and Rao et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 32 is respectfully requested.

6. Claim 33.

Dependent claim 33 recites that creating the checkpoint includes creating a new operation log file, and creating the checkpoint as a background operation.

Initially, claim 33 depends from claim 32. The disclosure of Rao et al. does not cure the deficiencies in the disclosures of Dinker et al. and Bobbitt et al. identified above with regard to claim 32. Therefore, claim 33 is patentable over Dinker et al., Bobbitt et al., and Rao et al., whether taken alone or in any reasonable combination, for at least the reasons given with regard to claim 32.

Further, Appellants submit that the Examiner has not established a prima facie case of obviousness with regard to claim 33. For example, the Examiner has not provided even a single reason why one of ordinary skill in the art, at the time of Appellants' invention, would have modified the Dinker et al. system to create a new operation log file and create a checkpoint as a background operation, as allegedly disclosed by Rao et al. Thus, the Examiner has not established a prima facie case of obviousness with regard to claim 33.

For at least these reasons, it is respectfully submitted that claim 33 is patentable over

Dinker et al., Bobbitt et al., and Rao et al., whether taken alone or in any reasonable combination, under 35 U.S.C. § 103. Reversal of the rejection of claim 33 is respectfully requested.

VIII. CONCLUSION

In view of the foregoing arguments, Appellant respectfully solicit the Honorable Board to reverse the Examiner's rejections of claims 1, 3-11, 13, 15, 16, and 19-39 under 35 U.S.C. § 103.

To the extent necessary, a petition for an extension of time under 37 C.F.R. § 1.136 is hereby made. Please charge any shortage in fees due in connection with the filing of this paper, including extension of time fees, to Deposit Account No. 50-1070 and please credit any excess fees to such deposit account.

Respectfully submitted,

HARRITY & HARRITY, LLP

/Paul A. Harrity, Reg. No. 39574/
Paul A. Harrity
Reg. No. 39,574

Date: May 6, 2009
11350 Random Hills Road
Suite 600
Fairfax, Virginia 22030
(571) 432-0800
Customer No. 44989

IX. CLAIM APPENDIX

1. A file system, comprising:

a plurality of servers configured to store file data as chunks; and

a master connected to the servers and configured to:

store namespace data that includes file identifiers for files for which the file data is stored as chunks,

store mapping data that maps the file identifiers to the chunks to which the file identifiers correspond,

store an operation log that includes a record of changes to at least one of the namespace data or the mapping data, and

store location data that identifies which of the servers stores which of the chunks, where the master is configured to:

communicate with the servers at startup of the master to identify the chunks stored by the servers, and

record, in a non-persistent manner, information regarding the chunks stored by each of the servers as the location data.

2. (canceled)

3. The system of claim 1, where the master is further configured to control placement of new chunks at the servers.

4. The system of claim 3, where when controlling the placement of new chunks, the master is configured to:

identify one or more of the servers to store the new chunks based on at least one of utilization of the servers, prior chunk distribution involving the servers, network topology, or failure correlation properties associated with the servers, and

place the new chunks at the identified one or more servers.

5. The system of claim 1, where the master is further configured to control redistribution of the chunks stored by the servers.

6. The system of claim 5, where when controlling redistribution of the chunks, the master is configured to:

select a chunk to redistribute based on a current distribution of the chunks,

identify one or more of the servers to which to move the selected chunk, and

move the selected chunk to the identified one or more servers.

7. The system of claim 1, where the master is further configured to monitor a state of the servers.

8. The system of claim 7, where the master is configured to exchange heartbeat signals with the servers to determine the state of the servers.

9. The system of claim 8, where the heartbeat signals include space utilization information.

10. The system of claim 7, where the state of the servers includes information regarding the chunks stored by the servers.

11. The system of claim 10, where the information includes version numbers of the chunks.

12. (canceled)

13. A master in a file system that includes the master connected to a plurality of servers, the master comprising:

means for communicating with the servers to identify file data stored by the servers as chunks;

means for storing, in a non-persistent manner, location information that identifies ones of the servers that store the chunks;

means for updating the location information by periodically instructing the servers to identify the data stored by the servers;

means for storing namespace data that includes file identifiers for files for which the file data is stored as chunks by the servers;

means for storing mapping data that maps the file identifiers to the chunks to which the

file identifiers correspond; and

means for storing an operation log that includes a record of changes to at least one of the namespace data or the mapping data.

14. (canceled)

15. A file system, comprising:

a plurality of servers configured to store files as chunks; and

a master connected to the servers and configured to:

store namespace data that includes file identifiers for which the files are stored as chunks,

store mapping data that maps the file identifiers to the chunks to which the file identifiers correspond,

store an operation log that includes a record of changes to at least one of the namespace data or the mapping data, and

store location data that identifies which of the servers stores which of the chunks, where the master is configured to:

determine location information by communicating with the servers, the location information being based on which of the servers store ones of the chunks,

store the location information as the location data, and

update the location data by periodically communicating with the servers to obtain changes to the location data.

16. A file system, comprising:

a plurality of servers configured to store file data as chunks; and

a master connected to the servers and configured to:

store namespace data that includes file identifiers for files for which the file data is stored as chunks,

store mapping data that maps the file identifiers to the chunks to which the file identifiers correspond,

store an operation log that includes a record of changes to the namespace data and the mapping data, and

store location data that identifies which of the servers stores which of the chunks, where the master is configured to:

communicate with the servers to determine location information of the data, the location information being based on which of the servers store the chunks, and

store the location information as the location data.

17. (canceled)

18. (canceled)

19. The file system of claim 1, where the file identifiers are organized hierarchically

in a tree of directories.

20. The file system of claim 1, where the master stores the namespace data using prefix-compression.

21. The file system of claim 1, where the master is configured to identify one of the chunks via a chunk handle that uniquely identifies the one of the chunks.

22. The file system of claim 21, where the chunk handle encodes a timestamp.

23. The file system of claim 1, where the master is configured to update the location data by periodically instructing the servers to provide information regarding the chunks stored by the servers.

24. The file system of claim 1, where the operation log includes a logical timeline that defines an order for concurrent operations.

25. The file system of claim 1, where the master is configured to:

determine when a size of the operation log exceeds a threshold, and

create a checkpoint of the operation log when the size of the operation log exceeds the threshold.

26. The file system of claim 25, where the master is configured to:

create a new operation log file, and

create the checkpoint as a background operation.

27. The master of claim 13, where the operation log includes a logical timeline that defines an order for concurrent operations.

28. The master of claim 13, further comprising:

means for determining when a size of the operation log exceeds a threshold; and

means for creating a checkpoint of the operation log when the size of the operation log exceeds the threshold.

29. The master of claim 28, where the means for creating the checkpoint includes:

means for creating a new operation log file, and

means for creating the checkpoint as a background operation.

30. A method performed by a master device in a file system that includes the master device connected to a plurality of server devices, the method comprising:

communicating with the server devices to identify file data stored by the server devices as chunks;

storing location information that identifies ones of the server devices that store the chunks;

storing namespace data that includes file identifiers for files for which the file data is stored as chunks by the server devices;

storing mapping data that maps the file identifiers to the chunks to which the file identifiers correspond; and

maintaining an operation log that includes a record of changes to the namespace data and the mapping data.

31. The method of claim 30, where maintaining the operation log includes storing a logical timeline that defines an order for operations including concurrent operations.

32. The method of claim 30, further comprising:

determining when a size of the operation log exceeds a threshold; and

creating a checkpoint of the operation log when the size of the operation log exceeds the threshold.

33. The method of claim 32, where creating the checkpoint includes:

creating a new operation log file, and

creating the checkpoint as a background operation.

34. The file system of claim 15, where the master is further configured to:

identify one or more of the servers to store a new chunk based on prior chunk distribution involving the servers, and

place the new chunk at the identified one or more servers.

35. The file system of claim 15, where the master is further configured to:

identify one or more of the servers to store a new chunk based on failure correlation properties associated with the servers, and

place the new chunk at the identified one or more servers.

36. The file system of claim 16, where the master is further configured to:

identify one or more of the servers to store a new chunk based on prior chunk distribution involving the servers, and

place the new chunk at the identified one or more servers.

37. The file system of claim 16, where the master is further configured to:

identify one or more of the servers to store a new chunk based on failure correlation properties associated with the servers, and

place the new chunk at the identified one or more servers.

38. The master of claim 13, further comprising:

means for identifying one or more of the servers to store a new chunk based on utilization of the servers, prior chunk distribution involving the servers, and failure correlation properties associated with the servers; and

means for placing the new chunk at the identified one or more servers.

39. The method of claim 30, further comprising:

identifying one or more of the server devices to store a new chunk based on utilization of the server devices, prior chunk distribution involving the server devices, and failure correlation properties associated with the server devices; and

placing the new chunk at the identified one or more server devices.

X. EVIDENCE APPENDIX

None

XI. RELATED PROCEEDINGS APPENDIX

None